REMARKS

Reconsideration of the present application is respectfully requested. Claims 48 and 51-62 have been canceled. Claims 44, 45 and 49 have been amended. No new matter has been added.

Drawing Objections

The Office objected to the drawings as failing to show the "plurality of bus registers" recited in the claims. Applicant respectfully submits that a "plurality of bus registers" are clearly shown in Figure 3 of the drawings, as the elements labeled with reference numerals 60, 61, 62, 64 and 66. Applicant therefore respectfully requests that the objection be withdrawn.

Specification Objections

The Office objected to the specification for several reasons, as discussed below. Note that a substitute specification (excluding the claims) is being submitted with this amendment, which reflects all changes made to the specification to date. No changes have been made to the specification in this amendment; therefore, only a clean version of the substitute specification is being provided.

The Office correctly pointed out that the amendment filed on 8/26/2002 misidentified a paragraph to be replaced as "the paragraph beginning at page 8, line 10." That paragraph should have been identified as "the paragraph beginning at page 8, line 5." This change is correctly reflected in the substitute specification filed herewith.

The Office also objected that the "plurality of bus registers" is not contained in the specification. Applicant respectfully submits that the "plurality of bus registers" is

disclosed in the substitute specification filed herewith in, for example, paragraph [0026] (registers 60, 61, 62, 64 and 66) and in subsequent paragraphs.

Applicant believes that the objections to the specification have been overcome.

Section 112 Rejections

The Office rejected claims 46, 47, 49 and 50 under 35 U.S.C. § 112, second paragraph as being indefinite. Specifically, the Office states that the term "vector processing assembly code level" in claim 46 is unclear. Applicant respectfully disagrees. The term "vector processing" is well-known in the field of microprocessor design, and its meaning is fully supported in Applicant's specification. "Vector processing" in this context refers to the ability of a single instruction to process an entire stream of data. In other words, a single instruction can automatically be repeated for a dynamical programmable number of times. This meaning is referred to in paragraph [0003] of the substitute specification and is also mentioned in the annex to the originally filed specification on page 22, middle of left column. The claims must be interpreted in light of Applicant's description. As such, the meaning of "vector processing assembly code level" in the claims is clear. Applicant respectfully submits, therefore, that this basis of rejection is incorrect and should be withdrawn.

The Office also stated that the claim term "plurality of bus registers" is unclear, stating that the specification makes no mention of a bus register. As indicated above, a "plurality of bus registers" is disclosed in the substitute specification in, for example, paragraph [0026] (registers 60, 61, 62, 64 and 66) and in subsequent paragraphs. Applicant respectfully submits, therefore, that this basis of rejection is incorrect and should be withdrawn.

Section 102 Rejections

Claims 44-50 were rejected under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent no. 4,766,566 of Chuang ("Chuang"). In addition, claims 44, 47 and 48 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent no. 5,487,022 of Simpson et al. ("Simpson").

Claim 44 has been amended to clarify what was meant by "free pipeline". Claim 44, as amended, recites:

> 44. (Currently amended) A processor comprising:
> a plurality of functional units coupled to each other to execute operations defined from an instruction set of the processor, the plurality of functional units including an arithmetic logic unit (ALU) and a multiplier, the instruction set having a hierarchy of instruction levels, each of which can be used by a programmer to define instructions for the processor, the hierarchy of instruction levels including:
> a RISC/CISC assembly code level, and
> a second assembly code level which includes **a plurality of instructions which are accessible to the programmer and which can explicitly reference individual outputs of any of the plurality of functional units**. (Emphasis added.)

The amendment to claim 44 essentially incorporates into claim 44 the limitations of dependent claim 48, which is now canceled.

Neither Chuang nor Simpson discloses or suggests a processor as recited in claim 44, or indicates why such a processor would be desirable. In particular, neither Chuang nor Simpson discloses or suggests a processor that has instructions accessible to the programmer which can explicitly reference individual outputs of any of a plurality of functional units, including an ALU and a multiplier, much less having such instructions in addition to a RISC/CISC assembly code level in a hierarchy of instruction levels that are accessible to the programmer.

The processors disclosed by Chuang and Simpson each comprises only a single level of instructions accessible to the programmer. The processor disclosed in Simpson has only a RISC instruction level. Chuang mentions the generally used method of microcode ROM to define more complex (CISC) instructions constructed by the elementary instructions which can be executed directly by the hardware (RISC). Note that that method is intended to define extra instructions, not to write programs, and the ROM is therefore limited to a few kBytes.

The processors disclosed by Chuang and Simpson do not have any instructions which are accessible to the programmer which can explicitly reference individual outputs of functional units, including an ALU and a multiplier (referred to as "free pipeline" in Applicant's specification and in the claims prior to this amendment). The processors disclosed by Chuang and Simpson both use a classical register file based instruction set, where the instructions which are accessible to the programmer refer to registers in the register file, not the outputs of individual functional units. In the classical register file based approach, data for an instruction is first loaded from the register file, then processed, and then the result of the instruction is written back to the register file, and all of these steps are undividably connected.

Among other advantages of the present invention, the recited "second assembly code level" ("free pipeline" assembly code level) enables the programmer to control all of those individual steps separately. With the processors disclosed by Chuang and Simpson, in contrast with the present invention, the programmer has no ability to route data freely through the various functional units. In addition, the present invention creates many more data paths through which data can move unrestrictedly from one

functional unit to the other without ever having to access the register file. This is an important advantage, because the register file tends to becomes a significant bottleneck in prior art processors that contain many functional units.

For at least the above reasons, therefore, claim 44 and all claims which depend on it are patentable over the cited art.

Dependent Claims

In view of the above remarks, a specific discussion of the dependent claims is considered to be unnecessary. Therefore, Applicants' silence regarding any dependent claim is not to be interpreted as agreement with, or acquiescence to, the rejection of such claim or as waiving any argument regarding that claim.

Conclusion

For the foregoing reasons, the present application is believed to be in condition for allowance, and such action is earnestly requested.

If any additional fee is required, please charge Deposit Account No. 02-2666.

Respectfully submitted,
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: 4/13/06

Jordan M. Becker
Reg. No. 39,602

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, CA 90025-1026
(408) 720-8300